REDUNDANT REPRESENTATION WITH COMPLEX WAVELETS: HOW TO ACHIEVE SPARSITY

Nick Kingsbury and Tanya Reeves

Signal Processing Group, Dept. of Engineering, University of Cambridge,
Cambridge CB2 1PZ, UK. E-mail: ngk@eng.cam.ac.uk

ABSTRACT

Overcomplete transforms, like the Dual-Tree Complex Wavelet Transform, offer more flexible signal representations than critically-sampled transforms, due to their properties of shift invariance and directional selectivity. We show that many transform coefficients can be discarded without much reconstruction quality loss by forcing compensatory changes in the remaining coefficients. We consider the convergence properties of an iterative projection system for achieving the usual coding aims of good sparsity with low reconstruction error. Results show how these measures translate to useful image compression performance.

1. INTRODUCTION

We have previously developed the Dual-Tree Complex Wavelet Transform (DT CWT) as a useful shift-invariant and directionally selective image analysis tool [1]. Here we consider how these properties may be harnessed for image compression, despite the DT CWT's 4:1 redundancy (overcompleteness). Matching Pursuits [2] is a well known technique for coding with overcomplete dictionaries, but it tends to be very computationally demanding. In this paper we discuss iterative projection techniques, introduced in [3], in order to achieve efficient coding with potentially lower computational costs. Bolcskei and Hlawatsch [4] have previously examined the effect of additive noise in oversampled filter banks systems, and Fischer and Cristobal [5] have proposed an iterative loop similar to ours.

When an overcomplete transform is employed, the inverse transform involves a projection from the higher dimensional transform space to the lower dimensional image space; e.g. from $4N$-space to N-space in the case of the 4:1 overcomplete DT CWT on an N-pixel image. Hence within the $4N$-space there is the N-dimensional range space and an orthogonal $3N$-dimensional null space of the transform. Movement within the range space produces changes in the output image whereas movement within the null space produces no change; and so different configurations of wavelet coefficients with the same range space component, but with different components in null space, can produce the same decoded output image.

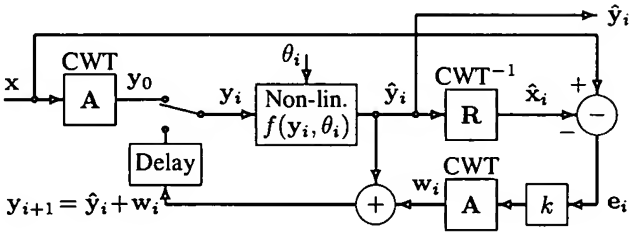

Fig. 1. Iterative-projection system block diagram. i is the iteration number.

For image coding purposes, we would like to find the configuration that concentrates image energy in as few non-zero wavelet coefficients as possible, while still producing a good approximation to the original image. The scheme presented here attempts iteratively to modify large coefficients to compensate as far as possible for the loss of small coefficients, by minimizing the error between the output image and the original.

2. TRANSFORM PROJECTIONS

Figure 1 shows the block diagram of our iterative algorithm. Let $\mathbf{x}$ be an N-vector representing the original N-pixel real-valued image in 1D-vectorized form, and let $\mathbf{A}$ be the $M \times N$ DT CWT analysis operator matrix, where $M = 4N$ and the rows of $\mathbf{A}$ are purely real and alternately represent the real and imaginary parts of the transform bases. Then $\mathbf{y} = \mathbf{A}\mathbf{x}$, where $\mathbf{y}$ is the M-vector of real and imaginary parts of the DT CWT coefficients. The real $N \times M$ synthesis or reconstruction operator matrix is $\mathbf{R}$, so that $\mathbf{x} = \mathbf{R}\mathbf{y}$. It is then easy to show that for a perfect reconstruction transform, $\mathbf{R}$ is given by:

$$\mathbf{R} = \hat{\mathbf{R}} + \mathbf{U}[\mathbf{I} - \mathbf{A}\hat{\mathbf{R}}] \quad \text{where } \hat{\mathbf{R}} = [\mathbf{A}^T\mathbf{A}]^{-1}\mathbf{A}^T \quad (1)$$

Hence $\mathbf{R}\mathbf{A} = \hat{\mathbf{R}}\mathbf{A} = \mathbf{I}$. Note that $\mathbf{U}$ is an arbitrary constant matrix defining a family of possible $\mathbf{R}$'s, $\mathbf{A}^T$ is the transpose of $\mathbf{A}$, and $\hat{\mathbf{R}}$ is the pseudo-inverse of $\mathbf{A}$. We shall assume that the energy of $\mathbf{R}$ is minimized, so that $\mathbf{U} = \mathbf{0}$ and $\mathbf{R} = \hat{\mathbf{R}}$, which is closely approximated in the DT CWT.

0-7803-7750-8/03/$17.00 ©2003 IEEE.

Let $\mathcal{S}$ be the space of all transform domain signals $\mathbf{y}$, obtainable from finite real input signals $\mathbf{x}$; i.e. $\mathcal{S}$ is the range space of $\mathbf{A}$. The projection operator onto $\mathcal{S}$ is $\mathbf{P}^{\mathcal{S}} = \mathbf{A}\hat{\mathbf{R}}$. If $\mathcal{S}^{\perp}$ is the orthogonal complement space of $\mathcal{S}$, the projection operator onto $\mathcal{S}^{\perp}$ is $\mathbf{P}^{\perp} = \mathbf{I} - \mathbf{A}\hat{\mathbf{R}}$ so that $\mathbf{y} = \mathbf{P}^{\mathcal{S}}\mathbf{y} + \mathbf{P}^{\perp}\mathbf{y}$ for any $\mathbf{y}$. From equ. (1) we see that $\hat{\mathbf{R}}\mathbf{P}^{\mathcal{S}} = \hat{\mathbf{R}}\mathbf{A}\hat{\mathbf{R}} = \hat{\mathbf{R}}$ and $\hat{\mathbf{R}}\mathbf{P}^{\perp} = \hat{\mathbf{R}} - \hat{\mathbf{R}}\mathbf{A}\hat{\mathbf{R}} = 0$. Hence $\mathbf{P}^{\mathcal{S}}\mathbf{y}$ is the range-space component of $\mathbf{y}$ which defines the output image $\mathbf{x} = \hat{\mathbf{R}}\mathbf{y}$, while $\mathbf{P}^{\perp}\mathbf{y}$ is the null-space component that has no effect on $\mathbf{x}$.

The non-linear function $f(\mathbf{y}, \theta)$ in fig. 1 is designed to suppress small amplitude coefficients, and optionally to quantize those of larger amplitude. The simplest useful function is a hard threshold centre-clipper, operating on the magnitudes of the *complex* coefficients to eliminate insignificant coefficients and leave the significant coefficients unaffected. It produces $\hat{\mathbf{y}} = f(\mathbf{y}, \theta)$, where the two components of the k^{th} complex element of $\hat{\mathbf{y}}$, for $k = 1 \ldots 2N$, are given by:

$$\hat{y}_{2k-1} + j\,\hat{y}_{2k} = \begin{cases} 0 & \text{if } |y_{2k-1} + j\,y_{2k}| < \theta \\ y_{2k-1} + j\,y_{2k} & \text{otherwise} \end{cases} \qquad (2)$$

When $i = 0$, $\mathbf{y}_i$ is initialized to $\mathbf{y}_0$, the transform of $\mathbf{x}$. The centre-clipper then sets all small coefficients of $\mathbf{y}_i$ to be zero in $\hat{\mathbf{y}}_i$. The error image $\mathbf{e}_i$ is the error between $\mathbf{x}$ and $\hat{\mathbf{x}}_i$, the inverse transform of the sparse vector $\hat{\mathbf{y}}_i$.

Assuming $k = 1$ for the moment, $\mathbf{w}_i$ is the transform of $\mathbf{e}_i$, and each wavelet coefficient in $\mathbf{w}_i$ defines the amount of each wavelet basis function present in the error image. These components tend to modify the non-zero coefficients of $\hat{\mathbf{y}}_0$ such that they are increased in amplitude while the coefficients of $\mathbf{y}_1$, which were set to zero in $\hat{\mathbf{y}}_0$, tend to be reduced compared with their amplitudes in $\mathbf{y}_0$. In this way, the error $\mathbf{e}_1$ after one iteration of the loop is significantly reduced compared with the initial error $\mathbf{e}_0$. Further iterations continue to reduce the error $\mathbf{e}_i$ until convergence occurs.

For a more rigorous analysis, we split $\mathbf{y}_{i+1}$ into its range-space and null-space components:

$$\begin{aligned} \mathbf{y}_{i+1} &= \hat{\mathbf{y}}_i + \mathbf{w}_i = \hat{\mathbf{y}}_i + k\mathbf{A}(\mathbf{x} - \hat{\mathbf{R}}\hat{\mathbf{y}}_i) \qquad (3) \\ &= \hat{\mathbf{y}}_i + k\mathbf{y}_0 - k\mathbf{P}^{\mathcal{S}}\hat{\mathbf{y}}_i = k\mathbf{y}_0 + (\mathbf{I} - k\mathbf{P}^{\mathcal{S}})\hat{\mathbf{y}}_i \\ &= \mathbf{y}_0 + \mathbf{P}^{\perp}\hat{\mathbf{y}}_i \quad \text{if } k = 1 \qquad (4) \end{aligned}$$

Hence, if $k = 1$, each new $\mathbf{y}$ equals the original $\mathbf{y}_0$ plus any null-space components of the previous $\hat{\mathbf{y}}$. Since the null space is orthogonal to the range space, the range space component $\mathbf{P}^{\mathcal{S}}\mathbf{y}_i = \mathbf{P}^{\mathcal{S}}\mathbf{y}_0 = \mathbf{y}_0$ for all i; and so the inverse transform of every $\mathbf{y}_i$ will be $\mathbf{x}$ (i.e. *every* $\mathbf{y}_i$ *is a valid representation of* $\mathbf{x}$ *in the transform domain*).

3. CONVERGENCE ANALYSIS

We may analyze convergence using the theory of Projection onto Convex Sets (POCS). Following Yang [6], a set is convex if any linear interpolation between any two members of the set is also in the set. The projection $P_i\mathbf{f}$ onto a set C_i from an arbitrary vector $\mathbf{f}$ finds the member of C_i that is closest to $\mathbf{f}$.

If the nonlinear function in fig. 1 is a centre-clipper, as defined in equ. (2), we may separate the clipping action into two steps: the first step selects which coefficients of $\mathbf{y}_i$ are to be retained, by generating a mask vector $\mathbf{m}_i$ of zeros and ones; the second step multiplies $\mathbf{y}_i$ by $\mathbf{m}_i$ (element-by-element) to give $\hat{\mathbf{y}}_i$. Given the mask $\mathbf{m}_i$, this second step is a projection P_1 from $\mathbf{y}_i$ onto the convex set C_1 of all vectors whose non-zero elements are those selected by ones in the mask. Hence $\hat{\mathbf{y}}_i = P_1\mathbf{y}_i$.

From equ. (4), the remaining parts of the loop in fig. 1 may be shown, if $k = 1$, to be a projection P_2 from $\hat{\mathbf{y}}_i$ to $\mathbf{y}_{i+1} = P_2\hat{\mathbf{y}}_i$, where the convex set C_2 is now the set of all vectors whose range-space component is $\mathbf{y}_0 = \mathbf{A}\mathbf{x}$. Hence our loop comprises repeated projections between C_1 and C_2, given by $\mathbf{y}_{i+1} = P_2P_1\mathbf{y}_i$.

If the choice of mask $\mathbf{m}_i$ were fixed for all iterations, then sets C_1 and C_2 would be constant and, by the theory of POCS, the loop would converge either to a point where the two sets overlap or to the closest pair of points in the two sets if they do not overlap (the more likely case for lossy compression). But $\mathbf{m}_i$ is not constant, so we invoke the following argument.

If we first choose $\mathbf{m}_0$ based on picking the largest M_{nz} complex coefficients from $\mathbf{y}_0$ as the non-zeros, and then iterate the POCS loop for L iterations with $\mathbf{m}_i = \mathbf{m}_0$ (i.e. with constant C_1) until $\hat{\mathbf{y}}_i$ approximately converges, we find the optimum $\hat{\mathbf{y}}_i$ given that $\mathbf{m}_i = \mathbf{m}_0$. At convergence, $\mathbf{y}_{i+1} \simeq \mathbf{y}_i$ and so the transform domain loop error is

$$\mathbf{y}_i - \hat{\mathbf{y}}_i \simeq \mathbf{y}_{i+1} - \hat{\mathbf{y}}_i = \mathbf{w}_i = \mathbf{A}\mathbf{e}_i \qquad (5)$$

which is entirely within $\mathcal{S}$. Now we can pick a better $\mathbf{m}_i = \mathbf{m}_L$, where $\mathbf{m}_L$ is based on selecting the largest M_{nz} coefficients from $\mathbf{y}_L$ as the non-zeros (as in a centre-clipper). This modifies C_1 so that the error $||\mathbf{y}_i - \hat{\mathbf{y}}_i||$ becomes smaller. This will result in a smaller image domain error $||\mathbf{e}_i||$ for two reasons: (a) because $\mathbf{e}_i = \mathbf{R}(\mathbf{y}_0 - \hat{\mathbf{y}}_i) = \mathbf{R}(\mathbf{y}_i - \hat{\mathbf{y}}_i)$; and (b) because the new $\mathbf{y}_i - \hat{\mathbf{y}}_i$ will probably no longer be in $\mathcal{S}$ and so $\mathbf{e}_i$, its projection into the image space, will be smaller still. Hence modifying the mask at regular intervals to simulate a centre-clipper, can produce further reductions in loop error, in addition to the reductions produced by POCS with a fixed mask.

From here we can argue that updating the mask on *every* iteration will produce more rapid convergence than updating it less frequently after waiting for the POCS to converge each time. This converts our two-step clipper back into a regular centre-clipper, and shows that the loop with centre-clip non-linearity will converge to a point which locally minimizes the image domain error $||\mathbf{e}_i||$. Our experiments support the validity of this argument.

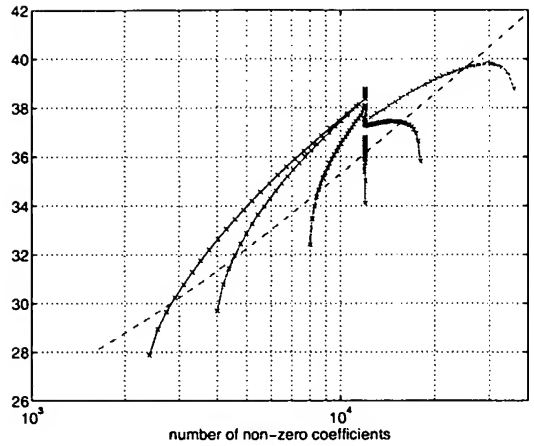

Fig. 2. Convergence of the PSNR (dB) vs number of non-zero complex coefficients M_{nz} for various increasing, constant and decreasing trajectories of M_{nz} with i, over 30 iterations in each case. The dashed line shows the rate-distortion curve for the non-redundant DWT, for comparison. (Note: the DWT coefficients are real, not complex.)

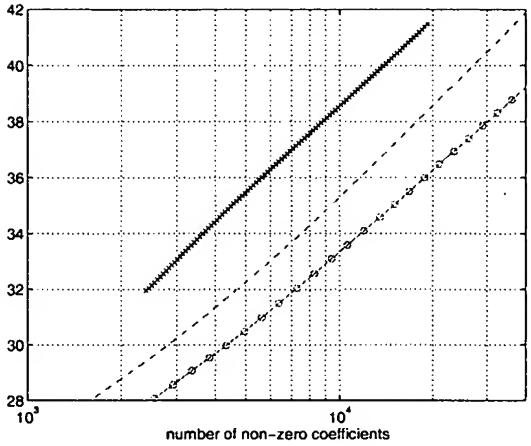

Fig. 3. PSNR (dB) vs. M_{nz} for the best iterative DT CWT scheme (upper curve), the non-iterated DWT (dashed curve) and the non-iterated DT CWT (lower curve).

4. CONVERGENCE EXPERIMENTS

We now turn to ways of encouraging the local minimum to be a good one. We consider the effect of adapting the sparsity of $\hat{\mathbf{y}}_i$ with iteration i.

First, note that there is *hysteresis* in the system. Suppose the clipping threshold is gradually being reduced. When a coefficient first exceeds the threshold, it suddenly appears in $\hat{\mathbf{y}}_i$. Subsequent iterations of the loop tend to increase the amplitude of this coefficient significantly above the threshold. We would then have to raise the clipping threshold to perhaps twice its original value before that coefficient would disappear from $\hat{\mathbf{y}}_i$ again. Hence there is hysteresis and the order in which coefficients are selected or removed during convergence becomes important.

Figure 2 illustrates a test of this by plotting PSNR ($= 10\log_{10}(255^2 N/||\mathbf{e}_i||^2)$ against number of non-zero coefficients M_{nz}. We use the 512×512 Lena image and 5 levels of wavelet decomposition. The six solid curves with crosses indicate how the PSNR varies with M_{nz}, starting from six different initial values of M_{nz} and all converging on a final desired $M_{nz} = 12000$. The initial values are 2400, 4000, 8000, 12000, 18000 and 36000; and in each case M_{nz} converges to 12000 over 26 iterations in a geometric series. Four further iterations are then performed with $M_{nz} = 12000$ to allow final convergence. We see that the curve with constant M_{nz} has the worst performance, and that it is better to increment M_{nz} from quite a low initial value than to decrement it from a high value.

Taking the best result (the left-hand curve), it is clear that considerable performance gains are possible compared with a non-iterated system, whose performance is shown by the initial points on each of the six curves. For example we see that the best converged result achieves almost the same PSNR with 12000 coefficients (38.79 dB) that the non-iterated DT CWT achieves with 36000 coefficients (38.77 dB). Looking vertically, we also see that the best converged result is 4.66 dB better than the non-iterated result of 34.11 dB.

There are several ways to obtain small improvements on the basic POCS loop, described above. Two of these are:

(a) There is loss around the loop due to both projection processes. We can compensate for this by increasing k. From equ. (4), range-space components of $\hat{\mathbf{y}}_i$ have a gain of $(1-k)$. For stability $(1-k)$ must lie inside the unit circle, so $0 < k < 2$. In practice, we use $k = 1.8$.

(b) The hysteresis of the system is caused by the very high gain of the centre-clipper characteristic around its threshold. To reduce this problem we replace the clipper with a similar non-linear function that has limited maximum slope, such as the Wiener denoising function:

$$\hat{y} = \begin{cases} 0 & \text{if } |y| < \theta \\ y \,.\, \dfrac{|y|^2 - \theta^2}{|y|^2} & \text{otherwise} \end{cases} \qquad (6)$$

This has zero gain below threshold and closely approximates unit gain when $|y| \gg \theta$, but it is continuous and has a maximum gradient of 2. If this function replaces the centre-clipper for the early iterations, then better patterns of non-zero coefficients are indeed produced and there is less need for the iterations to start with a very small value of M_{nz}.

These improvements contribute a small gain in final PSNR (typically 0.3 to 0.9 dB) and some improvement in rate of convergence. Figure 3 compares the performance of the best

iterated DT CWT scheme (for $k = 1.8$) over a range of M_{nz} values, with that of the DWT and DT CWT (both non-iterated). The iterated curve was produced by first using 15 iterations of Wiener non-linearity and then 15 iterations of centre-clipping to converge to a good starting point with $M_{nz} = 2400$. Then we incremented M_{nz} by about 2% on each of 100 further iterations to produce the 100 points in fig. 3. This figure seems to show a dramatic superiority for the iterated DT CWT scheme, *but* the complex DT CWT coefficients will need more bits to code each of them than the real DWT coefficients. However this will be much less than twice as many bits, because in a sparse data set the location of each non-zero coefficient often requires more bits to code it than the magnitude and sign (or phase) do.

5. CODING RESULTS

We now consider fully quantized systems (not just centre-clipped ones) and estimate the bit rate based on simple entropy measurements of the quantized data at each scale. Proper coding methods such as those used in SPIHT and JPEG2000 will give small improvements over simple entropy, but the *relative* performances of the energy compression and quantization processes, which are the main topic of this paper, should be little altered by this. For all the results in this paper we used the standard Daubechies (9,7)-tap filters in the DWT and the following filters in the DT CWT [1]: (13,19)-tap near-orthogonal filters at level 1, 14-tap Q-shift filters below level 1.

For these tests, we took the upper curve of fig. 3 and at every third point introduced a 2-D circularly symmetric complex quantizer into the loop in place of the centre clipper. The quantizer has Voronoi regions, made up of concentric rings of equal width ΔR around a circle of diameter $2\Delta R$, centred on the origin. The central circle forms the 'zero' bin of the quantizer and is undivided, while each ring is divided into approximately 'square' sectors for coding the phase. There are 8 sectors in ring 1, 12 in ring 2, 16 in ring 3 etc. (i.e. $4(k+1)$ sectors in the ring of inner radius $k\Delta R$). The real coefficients of the DWT are coded using the equivalent 1-D quantizer in which the central 'zero' bin is of width $2\Delta R$ and the other bins are all of width ΔR.

The first-order entropies of the quantized samples were measured at each wavelet scale, and from these the number of bits to code the Lena image were calculated over a range of step sizes. For the iterated DT CWT the average number of bits to code each non-zero complex coefficient is 13.5 when the PSNR is around 36 dB. For the DWT the average number of bits to code each non-zero real coefficient is 7.4 at the same PSNR. Figure 4 shows that, over the range 32 to 38 dB, the DT CWT scheme reduces the entropy by a factor of 0.86 (14%) at around 34 dB PSNR. This is equivalent to an improvement of 0.65 dB in PSNR for a given entropy.

Subjective comparisons of image quality indicate that

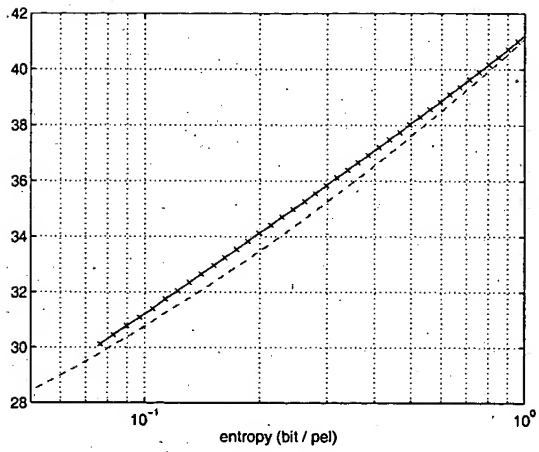

Fig. 4. PSNR (dB) vs. entropy (bit/pel) for 512×512 Lena, comparing coding of the complex coefficients of the iterated DT CWT (solid curve with crosses) with coding of the real coefficients of the non-iterated DWT (dashed curve).

coding of diagonal edges (such as those on Lena's hat) can be significantly improved with the DT CWT approach. The good directional selectivity of the DT CWT is clearly beneficial here. Further work is required to develop good coding schemes for the DT CWT (e.g. derivatives of SPIHT) which take full advantage of the spatial correlations in coding the locations of the non-zero coefficients. At present the number of iterations (typically about 30) to achieve good solutions is rather too high to make the proposed method attractive in real-time coding applications of reasonably large images, although the decoding process is non-iterative and very efficient.

6. REFERENCES

[1] N G Kingsbury, "Complex wavelets for shift invariant analysis and filtering of signals," *Applied and Computational Harmonic Analysis*, vol. 10, no. 3, pp. 234–253, May 2001.

[2] S G Mallat and Z Zhang, "Matching pursuits with time-frequency dictionaries," *IEEE Trans. Signal Proc.*, vol. 41, pp. 3397–3415, Dec. 1993.

[3] T H Reeves and N G Kingsbury, "Overcomplete image coding using iterative projection-based noise shaping," in *ICIP 02*, Rochester, Sept 2002, paper 2492.

[4] H Bölcskei and F Hlawatsch, "Oversampled filter banks: Optimal noise shaping, design freedom, and noise analysis," in *ICASSP 97*, April 1997, pp. 2453–2456.

[5] S Fischer and G Cristobal, "Minimum entropy transform using gabor wavelets for image compression," in *ICIAP'01*, Palermo, Sept 2001.

[6] Y Yang and N P Galatsanos, "Removal of compression artifacts using projections onto convex sets and line process modelling," *IEEE Trans. Image Proc.*, pp. 1345–1357, Oct. 1997.